Submission in Response to NSF CI 2030 Request for Information

DATE AND TIME: 2017-04-05 16:57:32
REFERENCE NO: 301

Author Names & Affiliations

- Johannes Dieterich - Mechanical and Aerospace Engineering, Princeton University

Contact Email Address (for NSF use only)

(Hidden)

Research Domain, discipline, and sub-discipline

sciences, computational chemistry, computational materials science, method development

Title of Submission

Cyberinfrastructure for Computational Chemistry – the importance of sustainable development practices

Abstract (maximum ~200 words).

Computational chemistry methods have become an important part of scientific inquiry. Their importance and scope will only increase in the future, requiring advanced cyberinfrastructure. However, beyond simply providing more and more raw computing power, we need to be able to properly exploit the resources now and in the future and continue to turn floating point operations into scientific insight. I argue, that sustainable development practices are key to ensuring this.

Question 1 Research Challenge(s) (maximum ~1200 words): Describe current or emerging science or engineering research challenge(s), providing context in terms of recent research activities and standing questions in the field.

Computational chemistry simulation methods have become a full partner of experiment, filling in knowledge gaps not easily accessible by experiment. The research challenges in our computational field have shifted from explaining experimental results by means of simple model systems to predicting material and molecular properties of complex systems at rapid throughput. The resulting dramatic increase of complexity can only be solved by better algorithms and accessible computing time in the future. However, scientists in the field typically employ a wild mix of codes, from commercial codes via open multi-institutional ones to smaller homegrown scripts and codes. Computing infrastructure then has the extraordinarily hard task to work (well) for all these different parts of the software stack, without necessarily realizing a priori what requirements they have in terms of required compilers, libraries, etc.

Question 2 Cyberinfrastructure Needed to Address the Research Challenge(s) (maximum ~1200 words): Describe any limitations or absence of existing cyberinfrastructure, and/or specific technical advancements in cyberinfrastructure (e.g. advanced computing, data infrastructure, software infrastructure, applications, networking, cybersecurity), that must be addressed to accomplish the identified research challenge(s).

As detailed before, the software ecosystem in computational chemistry is very diverse, ranging from codes scaling very well with the number of processing cores to memory-intensive, I/O intensive codes with very limited scalability. A two-pronged approach is and will be necessary: maintaining a similarly diverse computing infrastructure tailored for the different requirements on the one hand, while maintaining and extending legacy code bases and addressing their most severe limitations on the other.
The later obviously involves providing resources and training for researchers and scientists doing the actual development work in these code bases. New code bases must be designed to follow best practices in the field, ensuring their portability for future research generations. Nothing less than a cultural shift needs to happen, where scientists acquire a strong background in sustainable code development / best practices, and following them becomes mainstream. Promising steps in this direction have started to happen, more is needed.

Question 3 Other considerations (maximum ~1200 words, optional): Any other relevant aspects, such as organization, process, learning and workforce development, access, and sustainability, that need to be addressed; or any other issues that NSF should consider.

Personally, I am worried about ensuring portability of codes across software and hardware platforms and the need to subscribe to open computing standards and best practices. We must ensure maximal code sustainability over multiple generations of researchers as we have experienced that codes are used way beyond a single research project. E.g., in computational chemistry routines from the 1970ies are still in use in big commercial software package and will be for the foreseeable future. Modifications in or porting of these routines, speaking from experience, is incredibly difficult even if they contain only fully standard compliant code -- by 1970ies standards. The assumption then that relying on anything else than the best and most open, standard-compliant solutions in the year 2020 would not cause tremendous issues in 2060 is naive.
This however will require significant workforce training and another paradigm shift: moving from “how can we maximally exploit architecture X today” toward “how can we ensure the code runs well on X today and is easily portable in the future”. Some of the national leadership facilities emphasize this now but the most important facilities in the daily research of scientists – local resources – are in my experience not yet stressing this aspect enough. Knowledge among researchers of development strategies now commonly used in the technology industry to ensure code sustainability is very limited. Additionally, we may also need to acknowledge that the continuing existence of “PhD-ware” is promoted if not caused by the systemic emphasis on publishable scientific results over code development in academia.

Consent Statement